

IN THE CLAIMS:

A complete listing of the claims is set forth below. Please amend the claims as follows:

1. **(Currently Amended)** A computer-implemented system, comprising:

a data integration server coupled with one or more persistent data stores, the data integration server executing bulk data transfers between the one or more persistent data stores according to an enterprise-level business workflow, the data integration server comprising:

a plurality of programmatic source interfaces each coupled with one or more source data stores, defined according to a common programmatic source interface specification, and exposed within the data integration server during the bulk data transfer;

a plurality of relational interfaces used as alternatives to the plurality of programmatic source interfaces or the plurality of programmatic target interfaces based on file processing time and one or more performance requirements; and

a plurality of programmatic target interfaces, each coupled with one or more target data store, stores, defined according to a common programmatic target interface specification, and exposed within the data integration server during the bulk data transfer,

wherein each of the plurality of programmatic source interfaces extracts from the one or more source data stores one or more data entities for loading into any one or more target data stores during the bulk data transfer; and

wherein each of the plurality of programmatic target interfaces loads into the one or more target data stores the one or more data entities extracted from the one or more source data stores during the bulk data transfer.

2. **(Previously Presented)** The system of Claim 1, wherein the data integration server exposes bulk data transfer operations as services to applications or other systems within an enterprise-level infrastructure and executes a bulk data transfer operation in response to a request from such an application or other system.

3. **(Original)** The system of Claim 1, wherein the programmatic interfaces comprise JAVA interfaces.

4. **(Previously Presented)** The system of Claim 1, wherein:

a programmatic interface is exposed within the data integration server as an industry standard interface supporting bulk data transfers according to an industry standard protocol; and

the data integration server:

receives a request from a client indicating that the client is extracting data from or loading data into a data store in accordance with the industry standard protocol;

creates the programmatic interface to extract the data from or loading of the data into the data store; and

for data extraction, as the programmatic source interface produces the data extracted from the data store, sends the outgoing data to the client in accordance with the industry standard protocol; or

for data loading, as the data arrives from the client in accordance with the industry standard protocol, sends the incoming data to the programmatic target interface for loading into the data store.

5. **(Original)** The system of Claim 4, wherein the data integration server allows a client supporting an industry standard protocol for bulk data transfers to perform bulk data transfers with respect to an existing data store using a programmatic interface whether or not the existing data store or an associated existing application itself supports bulk data transfers in accordance with the industry standard protocol.

6. **(Previously Presented)** The system of Claim 1, wherein:

a programmatic source interface is exposed within the data integration server as an industry standard File Transfer Protocol (FTP) interface supporting bulk data transfers according to an FTP industry standard protocol; and

the data integration server:

allows an FTP client to open an FTP connection informing the data integration server that the FTP client is downloading a stream of data from the one or more source data stores;

creates the programmatic source interface to extract the stream of data from the one or more source data stores; and

as the programmatic source interface produces the stream of data extracted from the one or more source data stores, sends the outgoing stream of data to the FTP client in accordance with FTP.

7. **(Original)** The system of Claim 6, wherein at least one FTP client comprises a commercially available Extract-Transform-Load (ETL) tool supported within the data integration server.

8. **(Previously Presented)** The system of Claim 1, wherein:

a programmatic target interface is exposed within the data integration server as an industry standard File Transfer Protocol (FTP) interface supporting bulk data transfers according to an FTP industry standard protocol; and

the data integration server:

allows an FTP client to open an FTP connection informing the data integration server that the FTP client is uploading a stream of data to the one or more target data stores;

creates the programmatic target interface to enable loading of the stream of data into the one or more source data stores; and

as the stream of data arrives from the FTP client in accordance with FTP, sends the incoming stream of data to the programmatic target interface for loading into the one or more target data stores.

9. **(Original)** The system of Claim 8, wherein at least one FTP client comprises a commercially available Extract-Transform-Load (ETL) tool supported within the data integration server.

10. **(Previously Presented)** The system of Claim 1, wherein a particular data store is a source data store or a target data store for a particular bulk data transfer depending on whether data entities are extracted from the particular data store or loaded into the particular data store during the particular bulk data transfer.

11. **(Original)** The system of Claim 1, wherein loading data entities comprises inserting, updating, or deleting data entities.

12. **(Previously Presented)** The system of Claim 1, wherein:

each of the plurality of programmatic source interfaces and the plurality of programmatic target interfaces comprise one or more resources representing data entities contained in the one or more data stores are defined; and

the data integration server, in response to a request to execute a bulk data transfer involving one or more resources contained in one or more data stores, creates each programmatic interface within which at least one of the resources is defined.

13. **(Previously Presented)** The system of Claim 1, wherein the data integration server supports a commercially available Extract-Transform-Load (ETL) tool to:

connect directly to data stores with which the ETL tool is compatible to extract data entities directly from and load data entities directly into these data stores; and

connect to data stores, whether or not the ETL tool is compatible with these data stores, using the programmatic interfaces to extract data entities from and load data entities into these data stores.

14. **(Previously Presented)** The system of Claim 13, wherein the data integration server uses programmatic interfaces to support compatibility between any commercially available ETL tool and any one or more data stores.

15. **(Previously Presented)** The system of Claim 14, wherein the data integration server supports a controller to execute individual bulk data transfers using programmatic interfaces where either:

an Extract-Transform-Load (ETL) tool is not present; or

an ETL tool is present but its capabilities are not needed to transform data entities extracted from one or more source data stores, using one or more of the plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores, using one or more of the plurality of programmatic target interfaces, because physical database schemas of the source and target data stores are at least substantially similar.

16. **(Previously Presented)** The system of Claim 1, further comprising one or more transformation interfaces exposed within the data integration server, each transformation interface:

comprising one or more programmatic interfaces defined within the transformation interface;

comprising custom transformation logic applied to data entities extracted from one or more source data stores in a bulk data transfer, using one or more of the plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores in the bulk data transfer, using one or more of the plurality of programmatic target interfaces; and

isolating its custom transformation logic from its one or more defined programmatic interfaces; and

the data integration server further, in connection with creating the programmatic interfaces, creates each transformation interface within which at least one of the programmatic interfaces is defined for application of the associated custom transformation logic in the bulk data transfer.

17. **(Previously Presented)** The system of Claim 16, further comprising a controller supported within the data integration server to use a transformation interface in executing an individual bulk data transfer without using a commercially available Extract-Transform-Load (ETL) tool in connection with the bulk data transfer.

18. **(Currently Amended)** A computer-implemented method, comprising:

providing, by a server, a data integration server coupled with one or more persistent data stores, the data integration server executing bulk data transfers between the one or more persistent data stores according to an enterprise-level business workflow;

providing, by a server, a plurality of programmatic source interfaces, each coupled with one or more source data stores, defined according to a common programmatic source interface specification, and exposed within the data integration server during the bulk data transfer;

providing, by a server, a plurality of relational interfaces used as alternatives to the plurality of programmatic source interfaces or the plurality of programmatic target interfaces based on file processing time and one or more performance requirements;

extracting, by a server, from the one or more source data stores one or more data entities for loading into any one or more target data stores during the bulk data transfer;

providing, by a server, a plurality of programmatic target interfaces, each coupled with one or more target data stores, defined according to a common programmatic target interface specification, and exposed within the data integration server during a bulk data transfer; and

loading, by a server, into the one or more target data stores the one or more data entities extracted from any one or more source data stores during the bulk data transfer.

19. **(Previously Presented)** The method of Claim 18, wherein the data integration server exposes bulk data transfer operations as services to applications or other systems within an enterprise-level infrastructure and executes a bulk data transfer operation in response to a request from such an application or other system.

20. **(Original)** The method of Claim 18, wherein the programmatic interfaces comprise JAVA interfaces.

21. **(Previously Presented)** The method of Claim 18, wherein:

a programmatic interface is exposed within the data integration server as an industry standard interface supporting bulk data transfers according to an industry standard protocol; and

the method further comprises:

receiving a request from a client indicating that the client is extracting data from or loading data into a data store in accordance with the industry standard protocol;

creating the programmatic interface to extract the data from or loading of the data into the data store; and

for data extraction, as the programmatic source interface produces the data extracted from the data store, sending the outgoing data to the client in accordance with the industry standard protocol; or

for data loading, as the data arrives from the client in accordance with the industry standard protocol, sending the incoming data to the programmatic target interface for loading into the data store.

22. **(Original)** The method of Claim 21, further comprising allowing a client supporting an industry standard protocol for bulk data transfers to perform bulk data transfers with respect to an existing data store using a programmatic interface whether or not the existing data store or an associated existing application itself supports bulk data transfers in accordance with the industry standard protocol.

23. **(Previously Presented)** The method of Claim 18, wherein:

a programmatic source interface is exposed within the data integration server as an industry standard File Transfer Protocol (FTP) interface supporting bulk data transfers according to an FTP industry standard protocol; and

the method further comprises:

allowing an FTP client to open an FTP connection informing the data integration server that the FTP client is downloading a stream of data from the one or more source data stores;

creating the programmatic source interface to extract the stream of data from the one or more source data stores; and

as the programmatic source interface produces the stream of data extracted from the one or more source data stores, sending the outgoing stream of data to the FTP client in accordance with FTP.

24. **(Original)** The method of Claim 23, wherein at least one FTP client comprises a commercially available Extract-Transform-Load (ETL) tool supported within the data integration server.

25. **(Previously Presented)** The method of Claim 18, wherein:

a programmatic target interface is exposed within the data integration server as an industry standard File Transfer Protocol (FTP) interface supporting bulk data transfers according to an FTP industry standard protocol; and

the method further comprises:

allowing an FTP client to open an FTP connection informing the data integration server that the FTP client is uploading a stream of data to the one or more target data ~~store~~ stores;

creating the programmatic target interface to enable loading of the stream of data into the one or more source data stores; and

as the stream of data arrives from the FTP client in accordance with FTP, sending the incoming stream of data to the programmatic target interface for loading into the one or more target data stores.

26. **(Original)** The method of Claim 25, wherein at least one FTP client comprises a commercially available Extract-Transform-Load (ETL) tool supported within the data integration server.

27. **(Previously Presented)** The method of Claim 18, wherein a particular data store is a source data store or a target data store for a particular bulk data transfer depending on whether data entities are extracted from the particular data store or loaded into the particular data store during the particular bulk data transfer.

28. **(Original)** The method of Claim 18, wherein loading data entities comprises inserting, updating, or deleting data entities.

29. **(Previously Presented)** The method of Claim 18, wherein:

each of the plurality of programmatic source interfaces and the plurality of programmatic target interfaces comprise one or more resources representing data entities contained in the one or more data stores are defined; and

the method further comprises, in response to a request to execute a bulk data transfer involving one or more resources contained in one or more data stores, creating each programmatic interface within which at least one of the resources is defined.

30. **(Previously Presented)** The method of Claim 18, wherein the data integration server supports a commercially available Extract-Transform-Load (ETL) tool to:

connect directly to data stores with which the ETL tool is compatible to extract data entities directly from and load data entities directly into these data stores; and

connect to data stores, whether or not the ETL tool is compatible with these data stores, using the corresponding programmatic interfaces to extract data entities from and load data entities into these data stores.

31. **(Original)** The method of Claim 30, further comprising the data integration server using programmatic interfaces to support compatibility between any commercially available ETL tool and any corresponding data store.

32. **(Previously Presented)** The method of Claim 31, further comprising providing a controller within the data integration server to execute individual bulk data transfers using programmatic interfaces where either:

an Extract-Transform-Load (ETL) tool is not present; or

an ETL tool is present but its capabilities are not needed to transform data entities extracted from one or more source data stores, using one or more of the corresponding plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores, using one or more of the corresponding plurality of programmatic target interfaces, because physical database schemas of the source and target data stores are at least substantially similar.

33. **(Previously Presented)** The method of Claim 18, further comprising:

providing one or more transformation interfaces exposed within the data integration server, each transformation interface:

comprising one or more programmatic interfaces defined within the transformation interface;

comprising custom transformation logic applied to data entities extracted from one or more source data stores in a bulk data transfer, using one or more of the corresponding plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores in the bulk data transfer, using one or more of the corresponding plurality of programmatic target interfaces; and

isolating its custom transformation logic from its one or more defined programmatic interfaces; and

the data integration server, in connection with creating the programmatic interfaces, creating each transformation interface within which at least one of the programmatic interfaces is defined for application of the associated custom transformation logic in the bulk data transfer.

34. **(Previously Presented)** The method of Claim 33, further comprising providing a controller supported within the data integration server to use a transformation interface in executing an individual bulk data transfer without using a commercially available Extract-Transform-Load (ETL) tool in connection with the bulk data transfer.

35. **(Currently Amended)** A data integration server embodied with software executing bulk data transfers between persistent data stores according to an enterprise-level business workflow, the software when executed using one or more computers is configured to:

provide a data integration server coupled with one or more persistent data stores;

provide a plurality of programmatic source interfaces, each coupled with one or more source data stores, defined according to a common programmatic source interface specification, and exposed within the data integration server during the bulk data transfer;

provide a plurality of relational interfaces used as alternatives to the plurality of programmatic source interfaces or the plurality of programmatic target interfaces based on file processing time and one or more performance requirements;

extract from the one or more source data stores one or more data entities for loading into any one or more target data stores during the bulk data transfer;

provide a plurality of programmatic target interfaces, each coupled with one or more target data stores, defined according to a common programmatic target interface specification, and exposed within the data integration server during the bulk data transfer and

load into the one or more target data stores the one or more data entities extracted from any one or more source data stores during the bulk data transfer.

36. **(Previously Presented)** The data integration server of Claim 35, wherein the software is further configured to expose bulk data transfer operations of the data integration server as services to applications or other systems within an enterprise-level infrastructure and to execute a bulk data transfer operation in response to a request from such an application or other system.

37. **(Previously Presented)** The data integration server of Claim 35, wherein the programmatic interfaces comprise JAVA interfaces.

38. **(Previously Presented)** The data integration server of Claim 35, wherein:

a programmatic interface is exposed within the data integration server as an industry standard interface supporting bulk data transfers according to an industry standard protocol; and

the software is further configured to:

receive a request from a client indicating that the client is extracting data from or loading data into a data store in accordance with the industry standard protocol;

create the corresponding programmatic interface to extract the data from or loading of the data into the data store; and

for data extraction, as the programmatic source interface produces the data extracted from the data store, send the outgoing data to the client in accordance with the industry standard protocol; or

for data loading, as the data arrives from the client in accordance with the industry standard protocol, send the incoming data to the programmatic target interface for loading into the data store.

39. **(Previously Presented)** The data integration server of Claim 38, wherein the software is further configured to allow a client supporting an industry standard protocol for bulk data transfers to perform bulk data transfers with respect to an existing data store using a programmatic interface whether or not the existing data store or an associated existing application itself supports bulk data transfers in accordance with the industry standard protocol.

40. **(Previously Presented)** The data integration server of Claim 35, wherein:

a programmatic source interface is exposed within the data integration server as an industry standard File Transfer Protocol (FTP) interface supporting bulk data transfers according to an FTP industry standard protocol; and

the software is further configured to:

allow an FTP client to open an FTP connection informing the data integration server that the FTP client is downloading a stream of data from the corresponding source data store;

create the programmatic source interface to extract the stream of data from the corresponding source data store; and

as the programmatic source interface produces the stream of data extracted from the corresponding source data store, send the outgoing stream of data to the FTP client in accordance with FTP.

41. **(Previously Presented)** The data integration server of Claim 40, wherein at least one FTP client comprises a commercially available Extract-Transform-Load (ETL) tool supported within the data integration server.

42. **(Previously Presented)** The data integration server of Claim 35, wherein:

a programmatic target interface is exposed within the data integration server as an industry standard File Transfer Protocol (FTP) interface supporting bulk data transfers according to an FTP industry standard protocol; and

the software is further configured to:

allow an FTP client to open an FTP connection informing the data integration server that the FTP client is uploading a stream of data to the corresponding target data store;

create the programmatic target interface to enable loading of the stream of data into the corresponding source data store; and

as the stream of data arrives from the FTP client in accordance with FTP, send the incoming stream of data to the programmatic target interface for loading into the corresponding target data store.

43. **(Previously Presented)** The data integration server of Claim 42, wherein at least one FTP client comprises a commercially available Extract-Transform-Load (ETL) tool supported within the data integration server.

44. **(Previously Presented)** The data integration server of Claim 35, wherein a particular data store is a source data store or a target data store for a particular bulk data transfer depending on whether data entities are extracted from the particular data store or loaded into the particular data store during the particular bulk data transfer.

45. **(Previously Presented)** The data integration server of Claim 35, wherein loading data entities comprises inserting, updating, or deleting data entities.

46. **(Previously Presented)** The data integration server of Claim 35, wherein:

each of the plurality of programmatic source interfaces and the plurality of programmatic target interfaces comprise one or more resources representing data entities contained in the corresponding data store are defined; and

the software is further configured to, in response to a request to execute a bulk data transfer involving one or more resources contained in one or more data stores, create each programmatic interface within which at least one of the resources is defined.

47. **(Previously Presented)** The data integration server of Claim 35, wherein the software is further configured to support a commercially available Extract-Transform-Load (ETL) tool configured to:

connect directly to data stores with which the ETL tool is compatible to extract data entities directly from and load data entities directly into these data stores; and

connect to data stores, whether or not the ETL tool is compatible with these data stores, using the corresponding programmatic interfaces to extract data entities from and load data entities into these data stores.

48. **(Previously Presented)** The data integration server of Claim 47, wherein the ~~data~~ software is further configured to use programmatic interfaces to support compatibility between any commercially available ETL tool and any corresponding data store.

49. **(Previously Presented)** The data integration server of Claim 48, wherein the software is further configured to support a controller configured to execute individual bulk data transfers using programmatic interfaces where either:

an Extract-Transform-Load (ETL) tool is not present; or

an ETL tool is present but its capabilities are not needed to transform data entities extracted from one or more source data stores, using one or more of the corresponding plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores, using one or more of the corresponding plurality of programmatic target interfaces, because physical database schemas of the source and target data stores are at least substantially similar.

50. **(Previously Presented)** The data integration server of Claim 35, wherein the software is further configured to:

provide one or more transformation interfaces exposed within the data integration server, each transformation interface:

comprising one or more programmatic interfaces defined within the transformation interface;

comprising custom transformation logic applied to data entities extracted from one or more source data stores in a bulk data transfer, using one or more of the corresponding plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores in the bulk data transfer, using one or more of the corresponding plurality of programmatic target interfaces; and

isolating its custom transformation logic from its one or more defined programmatic interfaces; and

creating each transformation interface, in connection with creating the programmatic interfaces, within which at least one of the programmatic interfaces is defined for application of the associated custom transformation logic in the bulk data transfer.

51. **(Previously Presented)** The data integration server of Claim 50, wherein the software is further configured to provide a controller configured to use a transformation interface in executing an individual bulk data transfer without using a commercially available Extract-Transform-Load (ETL) tool in connection with the bulk data transfer.

52. **(Currently Amended)** A computer-implemented system, comprising:

a data integration server coupled with one or more persistent data stores, the data integration server executing bulk data transfers between the one or more persistent data stores according to an enterprise-level business workflow;

means for providing a plurality of programmatic source interfaces, each coupled with one or more source data stores, defined according to a common programmatic source interface specification, and exposed within the data integration server during the bulk data transfer;

means for providing a plurality of programmatic target interfaces, each coupled with one or more target data stores, defined according to a common programmatic target interface specification, and exposed within the data integration server during a bulk data transfer; and

means for providing a plurality of relational interfaces used as alternatives to the plurality of programmatic source interfaces or the plurality of programmatic target interfaces based on file processing time and one or more performance requirements;

wherein each of the plurality of programmatic source interfaces extracts from the one or more source data stores one or more data entities for loading into any one or more target data stores during the bulk data transfer; and

wherein each of the plurality of programmatic target interfaces loads into the one or more target data stores the one or more data entities extracted from the one or more source data stores during the bulk data transfer.

53. **(Currently Amended)** A computer-implemented system for executing bulk data transfers between persistent data stores according to an enterprise-level business workflow, comprising:

a data integration server coupled with one or more persistent data stores, the data integration server exposes its bulk data transfer operations as services to applications or other systems within an enterprise-level infrastructure and executes a bulk data transfer operation in response to a request from such an application or other system, the data integration server comprising:

a plurality of programmatic source interfaces, each associated with a corresponding source data store, defined according to a common programmatic source interface specification, and exposed within the data integration server during a bulk data transfer according to an enterprise-level business workflow to enable the data integration server to extract from the corresponding source data store one or more data entities for loading into any one or more selected target data stores during the bulk data transfer;

a plurality of programmatic target interfaces, each being associated with a corresponding target data store, defined according to a common programmatic target interface specification, and exposed within the data integration server during a bulk data transfer according to an enterprise-level business workflow to enable the data integration server to load into the corresponding target data store one or more data entities extracted from any one or more selected source data stores during the bulk data transfer;

a plurality of relational interfaces used as alternatives to the plurality of programmatic source interfaces or the plurality of programmatic target interfaces based on file processing time and one or more performance requirements;

wherein each of the plurality of programmatic source interfaces and the plurality of programmatic target interfaces:

provide to the corresponding source data store and the corresponding target data store an abstraction of bulk data transfer operations within the data integration server such that custom code need not be developed in connection with the corresponding source data

store and the corresponding target data store to enable bulk data transfers between the corresponding source data store and the corresponding target data store; and

isolate from the data integration server specific details associated with the corresponding source data store and the corresponding target data store such that custom code need not be developed in connection with the data integration server to enable bulk data transfers between the corresponding source data store and the corresponding target data store;

one or more transformation interfaces exposed within the data integration server, each transformation interface:

comprising one or more programmatic interfaces defined within the transformation interface;

comprising custom transformation logic applied to data entities extracted from one or more source data stores in a bulk data transfer, using one or more of the corresponding plurality of programmatic source interfaces, before the extracted data entities are loaded into one or more target data stores in the bulk data transfer, using one or more of the corresponding plurality of programmatic target interfaces; and

isolating its custom transformation logic from its one or more defined programmatic interfaces;

the data integration server further, in connection with creating the programmatic interfaces, create each transformation interface within which at least one of the programmatic interfaces is defined for application of the associated custom transformation logic in the bulk data transfer; and

a controller supported within the data integration server to use a transformation interface in executing an individual bulk data transfer without using a commercially available Extract-Transform-Load (ETL) tool in connection with the bulk data transfer.